Lab week 3-1: Heaps and HeapSort

Implement a Heap class to contain Integer

Eventually you may need to implement a Class **MyHeap (a min heap)** that contains arbitrary objects. But for this lab implement a min heap that contains integers. A MyHeap object will have a capacity. The capacity is the maximum number of items the heap can hold. It will also have a size. The size is the current number of objects in the heap

- Implement class MyHeap of integers that has:
  - Constructor creating an empty heap with default capacity = 50
    public MyHeap()
  - Constructor creating an empty heap with the capacity given as a parameter
    public MyHeap(int capacity)
  - Public method **buildHeap** that has a single explicit argument "**array of int**" and builds a heap using the **MyHeap object that is the implicit parameter**. It should return true if the build was successful and false if the capacity of the MyHeap object is not large enough to hold the "array of int" argument. **buildHeap must use** bottom up heap construction.
  - **boolean insert**(int) inserts the int argument into the heap.. Returns true if successful and false if not
  - int **findMin**(), returns the minimum value in the heap
  - int **deleteMin**(), deletes and returns the minimum value in the heap
  - boolean **isEmpty**(), returns true if the heap is empty, false otherwise
  - boolean **isFull**(), returns true if the number of items in the heap is equal to the capacity of the heap
  - Your build, insert, and delete methods should use one of public methods "void **driftDown**(int index)" or "void **driftUp(int index)**" . **Normally these would be private but make them public for testing purposes.**
  - Finally for testing purposes your MyHeap class should also contain the following:
    - **getHeapCap**() returns an integer that is maximum number of a entries **the heap** can hold.
    - **getHeapSize**() -- returns an integer that is the number of elements in the heap

Implement HeapSort

Implement Heap Sort as a method in the MyHeap class. At this point it will contain a single method with signature:

public static int[] heapSortDecreasing(int[])

This method should perform the sorting as discussed in class. It should build a heap from data passed. Thus the storage for the heap on which it is called will no longer be a heap. **It is not necessary to restore the heap to its original state.**

This takes an array of integers and returns an array with the integers in **decreasing** order. Note both the input array and output array begin their indexing at 0.

Submit to PolyLearn the classes: **MyHeap**